

**Gates & Cooper<sub>LLP</sub>**

Howard Hughes Center  
6701 Center Drive West, Suite 1050  
Los Angeles, California 90045

RECEIVED  
CENTRAL FAX CENTER

MAR 09 2005

**FAX TRANSMISSION TO USPTO**

TO: Commissioner for Patents  
Attn: Examiner Rachna Singh  
Patent Examining Corps  
Facsimile Center  
Alexandria, VA 22313-1450

FROM: George H. Gates  
OUR REF.: ARC9-98-157 (MCM)  
TELEPHONE: (310) 642-4146

Total pages, including cover letter: 11

PTO FAX NUMBER: (703) 872-9306

If you do NOT receive all of the pages, please telephone us at (310) 641-8797, or fax us at (310) 641-8798.

Title of Document Transmitted:	REPLY BRIEF OF APPELLANT.
Applicant:	Neelakantan Sundaresan
Serial No.:	09/191,281
Filed:	November 12, 1998
Group Art Unit:	2176
Title:	GENERATING VISUAL EDITORS FROM SCHEMA DESCRIPTIONS
Our Ref. No.:	ARC9-98-157 (MCM)

Please charge all fees to Deposit Account No. 09-0441 of IBM Corporation, the assignee of the present application.

By: GHG

Name: George H. Gates

Reg. No.: 33,500

I hereby certify that this paper is being transmitted by facsimile to the U.S. Patent and Trademark Office on the date shown below.

GHG  
Signature

3/9/05  
Date

GHG/sjm

G&C 30879.71-US-01

Confirmation No.: 8452  
Due Date: March 10, 2005**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE**

Applicant:	Neelakantan Sundaresan	Examiner:	Rachna Singh
Serial No.:	09/191,281	Group Art Unit:	2176
Filed:	November 12, 1998	Docket:	ARC9-98-157 (MCM)
Title:	GENERATING VISUAL EDITORS FROM SCHEMA DESCRIPTIONS		

**CERTIFICATE OF MAILING OR TRANSMISSION UNDER 37 CFR 1.8**I hereby certify that this correspondence is being filed via facsimile transmission to the U.S. Patent and Trademark Office on March 9, 2005.By: 

Name: George H. Gates

**MAIL STOP APPEAL BRIEF - PATENTS**

Commissioner for Patents

P.O. Box 1450

Alexandria, VA 22313-1450

Dear Sir:

We are transmitting herewith the attached:

- ☒ Transmittal sheet, in duplicate, containing a Certificate of Mailing or Transmission under 37 CFR 1.8.  
☒ REPLY BRIEF of Appellant(s).

Please consider this a PETITION FOR EXTENSION OF TIME for a sufficient number of months to enter these papers, if appropriate.

Please charge all fees to Deposit Account No. 09-0441 of IBM Corporation, the assignee of the present application. A duplicate of this paper is enclosed.

Customer Number 22462GATES & COOPER LLPHoward Hughes Center  
6701 Center Drive West, Suite 1050  
Los Angeles, CA 90045  
(310) 641-8797By: 

Name: George H. Gates

Reg. No.: 33,500

GHG/sjm

Due Date: March 10, 2005

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE**  
**BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES**

In re Application of:	)	
	)	
Inventor: Neelakantan Sundaresan	)	Examiner: Rachna Singh
	)	
Serial #: 09/191,281	)	Group Art Unit: 2176
	)	
Filed: November 12, 1998	)	Appeal No.: _____
	)	
Title: GENERATING VISUAL EDITORS	)	
<u>FROM SCHEMA DESCRIPTIONS</u>	)	

**RECEIVED**  
**CENTRAL FAX CENTER**

**REPLY BRIEF OF APPELLANT**

**MAR 09 2005**

Commissioner for Patents  
P. O. Box 1450  
Alexandria, Virginia 22313-1450

Dear Sir:

**I. Introduction**

In accordance with 37 CFR §41.37, Appellant's attorney hereby submits the Reply Brief of Appellant, in response to the Examiner's Answer dated January 10, 2005 in the above-identified application.

This Reply Brief of Appellant incorporates by reference herein the entirety of the previously filed Brief of the Appellant.

No fee is required for filing this Reply Brief of Appellant. However, the Office is authorized to charge any necessary fees or credit any overpayments to Deposit Account No. 09-0441 of IBM Corporation, the assignee of the present invention.

**II. Arguments**

**A. Appellant's Independent Claims Are Patentable Over The References**

In the Answer, the Examiner first restates the rejections of the independent claims from the Office Action. Thereafter, in the "Response to Argument," beginning on page 11, the Examiner responds to the arguments for the patentability of the independent claims set forth in the Brief of the Appellant as follows:

(11) Response to Argument

In reference to claims 1, 24, and 47, Appellant argues that the complete limitation, "generating one or more class specifications in the computer from a schema in a document, wherein the class specifications identify user interface components of the editor corresponding to entities define in the schema" is not shown by Dale. Examiner disagrees. As stated above, Dale teaches creating one director component in which the components are Java components embedded in the document. See column 10, lines 50-59. Compare to "generating class specifications in the computer system".

Dale also teaches displaying a graphical user interface on the display device and using an editor to integrate components into a hypertext pages to create an application. Dale's editor allows text to be combined with various components. See abstract and columns 19-20. Appellant argues that the editor in Dale merely constructs web-based applications, wherein the web pages include tags to applets that may be downloaded when the pages are accessed, and the director component is merely an applet interconnects other components. Examiner disagrees. Dale explicitly teaches creating a director component and displaying a graphical user interface (GUI) on a display device and using an editor to integrate components into a hypertext page to create an application. See abstract. The "integration of components in an editor" teaches Appellant's claimed invention because it teaches that class specifications ((director components) identify user interface components in the editor. Examiner emphasizes that the limitation of "identifying user interface components of the editor" is taught by Dale in his disclosure of using an editor to integrate components (class specifications). See abstract and columns 19-20.

Appellant argues that Dale's editor is not generated from class specifications that are themselves generated from a schema for a document. Examiner states in rejections, "It is noted that Dale does not explicitly say the components are "class specifications". However, refer to Kyojima's abstract and columns 1-4, in which he discloses the generation of "class specifications" from "schemas". It would have been obvious to one of ordinary skill in the art at the time of the invention to combine the inventions of Dale and Kyojima in arriving at the instant invention because such combination would facilitate adaptation of one type of component model to another according to structural constraints. See Kyojima's column 3, bottom in which he recites, "There is provided a document structure composing apparatus for composing document structure which meets the structural constraint of a predetermined document class and characterized in that complementation specification storage means for storing the element type of component which is required to complement a pre-complemented document structure composed according to a procedure, which does not meet the structure constraint of a first class, to adapt it to the first document class, and for storing a specification of complementation correlating to the element type and complementation means for analyzing said pre-complemented document structure based upon the structural constraint of the first document class. . ." Kyojima also discusses in the 'Background of the Invention' that one of the problems in the art at the time of the invention was that complementation in an area in document structure has an effect upon whether another area in document structure is suitable for the document class. The complementation to be performed in document structure not based upon only a

specific local area but based upon the structure of the whole document. The second problem is that a user has no way to select one method of plural methods of complementing which are suitable for the document class. Thus Kyojima's invention is made to solve problems to provide a document structure composing apparatus to meet structural constraints of a predetermined document class and characterized in that complementation specification storage means for storing the element type of component which is required to complement a pre-complemented document structure composed according to a procedure, which does not meet the structure constraint of a first class, to adapt it to the first document class, and for storing a specification of complementation correlating to the element type and complementation means for analyzing said pre-complemented document structure based upon the structural constraint of the first document class."

Appellant argues that Kyojima does not teach or suggest class specifications generated from a schema. Examiner respectfully disagrees. As indicated in the rejections and comments above, Kyojima discloses the generation of "class specifications" from "schemas". See Kyojima's abstract and columns 1-4. It would have been obvious to one of ordinary skill in the art at the time of the invention to combine the inventions of Dale and Kyojima in arriving at the instant invention because such combination would facilitate adaptation of one type of component model to another according to structural constraints. See Kyojima's column 3, bottom in which he recites, "There is provided a document structure composing apparatus for composing document structure which meets the structural constraint of a predetermined document class and characterized in that complementation specification storage means for storing the element type of component which is required to complement a pre-complemented document structure composed according to a procedure, which does not meet the structure constraint of a first class, to adapt it to the first document class, and for storing a specification of complementation correlating to the element type and complementation means for analyzing said pre-complemented document structure based upon the structural constraint of the first document class. . ."

Appellant argues that Dale only describes components that are applets and does not teach that the user interface components of the editor are instantiated as objects to invoke the editor. Dale teaches that the editor can be used to integrate selected components into an extended hypertext page. In Dale's system a connection between components of a set of hypertext pages is established and each of the components is referenced by at least one set of the pages.

Appellant's attorney disagrees with this analysis.

With regard to the limitation "generating one or more class specifications in the computer from a schema for the document, wherein the class specifications identify user interface components of the editor corresponding to entities defined in the schema," the Examiner cites Dale at col. 10, lines 50-59, because Dale teaches creating one director component in which the components are Java components embedded in the document. However, at the location indicated by the Examiner, the editor in Dale merely constructs web-based applications, wherein the web pages include tags to

applets that may be downloaded when the pages are accessed, and the director component is merely an applet interconnects other components (applets). There is no correspondence between this description in Dale and Appellant's claim limitation.

Also with regard to the limitation "generating one or more class specifications in the computer from a schema for the document, wherein the class specifications identify user interface components of the editor corresponding to entities defined in the schema," the Examiner cites Dale as teaching displaying a graphical user interface and using an editor to integrate components into a hypertext page to create an application, and that Dale's editor allows text to be combined with various components, at the Abstract and columns 19-20. However, Dale's editor, which is used to construct web-based applications including web pages having references or tags to applets, is not generated from class specifications that are themselves generated from a schema for the document, wherein the class specifications identify user interface components of the editor corresponding to entities defined in the schema.

With regard to the limitation "instantiating one or more objects in the computer from the class specifications to invoke the editor," the Examiner cites Dale at col. 10, lines 60-63 and col. 5, lines 7-9, as instantiating components, particularly Java components. However, Dale does not perform such instantiation in the context where the class specifications are generated from a schema for the document and identify user interface components of the editor corresponding to entities defined in the schema.

The Examiner admits that Dale does not explicitly state that the components are class specifications, but cites Kyojima, at the Abstract and cols. 1-4, as disclosing the generation of "class specifications" from "schemas." However, Kyojima merely describes document structure conversion, wherein the term class is used to refer to a type of document structure. Nowhere does Kyojima teach or suggest class specifications generated from a schema for a document, in the context where the class specifications identify user interface components of the editor corresponding to entities defined in the schema and are instantiated as objects to invoke the editor.

The Examiner asserts that that Kyojima teaches the limitations "generating class specifications from schemas," and that it would have been obvious to combine Dale and Kyojima, when it discloses a document structure composing apparatus for composing document structure which meets the structural constraint of a predetermined document class. However, this use of the term document class does not teach or suggest class specifications (in an object oriented programming sense) that generated from a schema for a document, in the context where the class

specifications identify user interface components of the editor corresponding to entities defined in the schema and are instantiated as objects to invoke the editor. Instead, Kyojima merely describes document structure conversion, wherein the term class is used to refer to a type of document structure.

The Examiner asserts that that Dale teaches the limitations "wherein the class specifications identify user interface components of the editor corresponding to entities defined in the schema," when it discloses that the editor can be used to integrate selected components into an extended hypertext page, that a connection between components of a set of hypertext pages is established, and that each of the components is referenced by at least one set of the pages. However, nowhere does Dale teach or suggest class specifications generated from a schema for a document, in the context where the class specifications identify user interface components of the editor corresponding to entities defined in the schema and are instantiated as objects to invoke the editor. Instead, Dale merely describes components that are applets, which are embodied as Java classes and embedded in HTML pages.

Consequently, Appellant's attorney respectfully submits that the combination of Dale and Kyojima does not render obvious the limitations of Appellant's independent claims. Moreover, the other references fail to overcome the deficiencies of Dale and Kyojima. Recall that the other references were only cited against dependent claims 2-3, 8-23, 25-26, 31-46, 48-49 and 54-69, in combination with Dale and Kyojima.

Appellant's attorney asserts that the differences between Appellant's claimed invention and the cited references result in operational advantages of the Appellant's invention over the cited references. In addition, Appellant's invention solves problems not recognized by the cited references.

Thus, Appellant's attorney submits that independent claims 1, 24 and 47 are allowable over the cited references.

#### B. Appellant's Dependent Claims Are Patentable Over The References

Dependent claims 2-23, 25-46 and 48-69 are submitted to be allowable over the cited references in the same manner, because they are dependent on independent claims 1, 24 and 47, respectively, and thus contain all the limitations of the independent claims. In addition, dependent claims 2-23, 25-46 and 48-69 recite additional novel elements not shown by the cited references. In

this regard, this Reply Brief of Appellant incorporates by reference herein the entirety of the previously filed Brief of the Appellant.

As noted above, in the Answer, the Examiner first restates the rejections of the dependent claims from the Office Action. Thereafter, in the "Response to Argument," beginning on page 15, the Examiner responds to some of the arguments for the patentability of the dependent claims set forth in the Brief of the Appellant as follows:

Appellant generally argues with regards to the dependent claims that limitations are not taught in the same context as Appellant's claimed invention. Examiner disagrees with respect to the rejections provided above. All of the Applicant's limitations with regard to the dependent claims are taught by the references. See rejection.

Appellant argues with regards to claims 5, 28, and 51 that Dale does not teach converting an entity defined in the schema into a class specification. Dale discloses the CPU executing instructions which convert received instructions, already known to be Java components, to instructions which can be directly executed by the CPU, which is known to be the instantiated objects from the class specification. See column 3, lines 64-67.

Appellant argues with respect to claims 6, 29, and 52 that Dale does not teach generation class specifications from the schemas and one or more optional customization specifications. Examiner disagrees. Dale discloses in column 20, 5-15 that the user can customize and select which components are included in the application which are "optional customization specifications."

Appellant argues with respect to claims 7, 30, and 53 that Dale does not teach the optional customization specifications define what class names to generate for each entity. Examiner disagrees. Dale discloses custom specifications that define the class names to generate, as in column 6, lines 18-20, where the programs are like applets where the class names must be defined, as in the example in column 14, lines 57-60.

Appellant argues with respect to claims 8, 31, and 54 that HotMetalPro does not teach that the class specifications include one or more specifications from a group of a visual editor class, a content implementation class, and a handler class. While Dale and Kyojima's system does not state a group comprising a visual editor class, a content implementation class, and handler class, HotMetalPro allows the user to use Java in the document editing window and provide various applets for visual means as well as customizing elements. See pages 77-80. It would have been obvious to one of ordinary skill in the art at the time of the invention to provide any of the above mentioned groups to the editor of Dale and Kyojima since a class specification could relate to the visual editor, the components of XML schemas and initiators of the visual editor.

Appellant argues with respect to claims 9, 32, and 55 that HotMetalPro does not teach mapping entities defined in the schema to components in the editor. HotMetal Pro teaches defining the applet with attributes and parameters. The Parameters can indicate the value and name of the object. See pages 78-79. It would have been obvious to one of ordinary skill in the art at the time of the invention to



map and select the elements and attribute of elements to the editor since the system of Dale and Kyojima and HotMetal Pro are concerned with providing class specifications to identify components of the editor.

Appellant argues with regards to claims 16, 39, and 62 that W3C does not teach that the class specifications include a function for validating at least one entity defined in the schema. Examiner disagrees. W3C teaches that there are validating processors that read the DTD and parse entities referenced in the document. See page 31 of W3C Recommendation XML 1.0.

Appellant argues with regards to claims 17, 40, and 63 that Bray does not teach generating class specifications from a regular expression language. Examiner disagrees for reasons stated in the rejection above.

Appellant argues with regards to claims 18-19, 41-42, and 64-65 that regular expression language is not taught by Bray. Examiner disagrees for reasons stated in the rejection above.

Appellant's arguments with respect to claims 20-23, 43-46, and 66-69 have been addressed in the rejection above.

Appellant's attorney disagrees.

With regard to claims 5, 28 and 51, which recite that the generating further comprises converting an entity defined in the schema into the class specification, the Office Action asserts that these limitations are taught by Dale at col. 3, lines 64-67. However, at the indicated location, Dale merely describes the execution of instructions by a CPU.

With regard to claims 6, 29 and 52, which recite that the generating further comprises generating the class specifications in the computer from the schemas and one or more optional customization specifications, the Office Action asserts that these limitations are taught by Dale at col. 20, lines 5-15. However, at the indicated location, Dale merely describes an HTML page with executable components.

With regard to claims 7, 30 and 53, which recite that the optional customization specifications define what class names to generate for each entity defined in the schema, the Office Action asserts that these limitations are taught by Dale at col. 6, lines 18-20 and col. 14, lines 57-60. However, at the indicated locations, Dale merely describes how "applets" are small portions of executable code that can be incorporated (by reference) into HTML pages, and an applet definition.

With regard to claims 8, 31 and 54, which recite that the class specifications include one or more specifications selected from a group comprising (1) a visual editor class specification, (2) a content implementation class specification, and a handler class specification, the Office Action asserts that these limitations are taught by HotMetalPro at pages 77-80. However, at the indicated location, HotMetalPro merely describes inserting applets in an HTML page.

With regard to claims 9, 32 and 55, which recite mapping the entities defined in the schema to components of the editor, the Office Action asserts that these limitations are taught by HotMetalPro at pages 78-79. However, at the indicated location, HotMetalPro merely describes the specification of applets.

With regard to claims 16, 39 and 62, which recite that the class specifications include at least one function for validating at least one entity defined in the schema, the Office Action asserts that these limitations are taught by W3C at page 31. However, at the indicated location, W3C merely describes validating processors for XML documents.

With regard to claims 17, 40 and 63, which recite that the generating further comprises generating the class specifications from a regular expression language comprising one or more declarations of elements enclosed within an element, the Office Action asserts that these limitations are taught by Bray at page 5 and 38. However, at the indicated location, Bray merely describes the grammar of XML.

### III. Conclusion

In light of the above arguments, Appellant's attorney respectfully submits that the cited reference does not anticipate nor render obvious the claimed invention. More specifically, Appellant's claims recite novel functions and features, which patentably distinguish over any and all references under 35 U.S.C. §§ 102 and 103.

As a result, a decision by the Board of Patent Appeals and Interferences reversing the Examiner and directing allowance of the pending claims in the subject application is respectfully solicited.

Respectfully submitted,

Appellant's attorneys,

GATES & COOPER LLP

Howard Hughes Center  
6701 Center Drive West, Suite 1050  
Los Angeles, California 90045  
(310) 641-8797

Date: March 9, 2005

By: George H. Gates  
Name: George H. Gates  
Reg. No.: 33,500

GHG/